300-120 2306

B-2750 FWC / 615179-4 #14
S.S.K.
1-22-97

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant: SAU-GEE CHEN and CHIEH-CHIH LI

Serial No.: 08/510,740

Filed: August 2, 1995

Title: METHOD FOR FINDING QUOTIENT IN A DIGITAL SYSTEM

Examiner: E. Moise

Art Unit: 2306

Re: APPELLANT'S BRIEF

Our Ref.: B-2750 FWC 615179-4

Date: December 20, 1996

I hereby certify that this correspondence is being deposited with the United States Postal Service as FIRST CLASS MAIL in an envelope addressed to: Commissioner of Patents and Trademarks, Washington, D.C. 20231, on:

December 20, 1996
Date of Mailing

Signature

APPEAL BRIEF

Honorable Commissioner of Patents and Trademarks
Washington, D.C. 20231

Sir:

The present application is under appeal to the Board of Patent Appeals and Interferences, the Notice of Appeal having been mailed on October 21, 1996.

A. STATUS OF CLAIMS

The present application, USSN 08/510,740, was filed on August 2, 1995, and is a file wrapper continuation application of USSN 08/188,068, filed on January 26, 1994.

The originally filed application contained Claims 1-2. In an amendment filed on December 20, 1994, in response to an Office Action dated September 2, 1994, Claims 1-2 were amended and additional Claims 3-4 were added. Claims 1-4 have not been amended since this date (through the remainder of prosecution of application USSN 08/510,740 and through the full prosecution history of file wrapper continuation USSN 08/188,068).

It is these Claims 1-4 filed with the response of December 20, 1994, (Claims 1-2 in an amended form and Claims 3-4 unchanged as filed), which form the subject of this appeal. The claims under appeal are attached hereto in the Appendix. The present status of the claims is as follows:

Claim 1 - pending; rejected under 35 USC 101.

Claim 2 - pending; rejected under 35 USC 101.

Claim 3 - pending; rejected under 35 USC 101.

Claim 4 - pending; rejected under 35 USC 101.

B. STATUS OF AMENDMENTS

No amendments have been made after final rejection of file wrapper continuation USSN 08/188,068, nor were any amendments made after final rejection of parent USSN 08/510,740.

C. SUMMARY OF THE INVENTION

1. Summary - Background and Narrative

The invention describes a novel digital information processing system employing a method for finding a quotient from a divisor and a dividend to enable the digital information processing system to operate in a faster and more efficient manner. A novel method for generating such quotient data in a digital information processing system is also described.

In a digital information processing system (such as a computer), division operations are inherently much slower than multiplication operations. This is because division operations in these systems are sequential. In the computer sciences, since division operations represent the slowest of all the basic mathematical operations (addition, subtraction, multiplication and division), efforts have been made since the advent of digital information processing systems to speed up division operations. (See Specification, Page 1, line 12 to line 17).

For instance, early attempts to speed up digital information processing systems eliminated the restoring operations of the partial remainders during a division operation or confined binary quotient digits to be either 1 or -1. However, the systems employing these improvements still needed to detect the sign of the remainders in the division operations. This requirement limited the advantages these improvements offered. (See Specification, Page 1, line 17 to Page 2, line 1).

More recently, algorithms utilizing SD (signed-digit) number representation have been applied to computer systems. Though considerably shortening the time required by computer systems for the remainder subtraction part of calculating a quotient, the increased complexity of the SD algorithm limited the advantages

that these improvements offered. Furthermore, the fact that the quotient resulting from these algorithms had to be converted to binary form by the computer system after being calculated slowed the division operations, thereby also limiting the advantages of computer systems using SD algorithms. (See Specification, Page 2, line 7 to line 22).

Other computer systems (especially multiplier-based computer processors) utilized division algorithms which essentially converted the division operation performed by the computer to a series of multiplication operations. Once again, computer systems employing these algorithms were still limited to slow iterative type operations and also requiring many more shift-and-add operations when calculating quotients. Therefore, these computer systems were also burdened with speed limitations in calculating quotients. (See Specification, Page 2, line 24 to Page 3, line 15).

Finally, a computer system employing an on-line division algorithm as described by K. S. Trivedi and M. D. Ercegovac in "On-Line Algorithms for Division and Multiplication," (IEEE Trans. On Computers, Vol. C-26, No. 7, July 1977), while presenting speed advantages for computers performing division operations, is also presented with significant speed limitations. This algorithm requires a complex three-input SD addition operation and requires more complex quotient decision circuitry. Further, a computer employing this algorithm must convert the quotient results into binary form. (See Specification, Page 3, line 17 to Page 4, line 4).

The present invention describes a digital information processing system and method in a digital processing system for generating faster quotient calculations than has heretofore been available. Specifically, to create a faster digital information processing system, the present invention calls for a digital processing system implementing a division algorithm which separates the remainder SD operation from the remainder subtraction operation in its division operations.

Heretofore, no computer system has implemented any such algorithm to achieve faster quotient processing operations. (See Specification, Page 4, line 10 to line 19).

In addition to utilizing processing operations whereby remainder SD operations and remainder subtraction operations are conducted separately and in parallel (speeding up quotient operations in the subject digital system), the applied algorithm employs the streamlined and time-saving non-restoring division operations and conventional carry-save-adder (CSA) type operations. Furthermore, use of a limited quotient digit set ({-1, 1}) in this on-line algorithm employed by the digital information processing system assures fast quotient calculation and conversion to binary form. (See Specification, Page 4, line 25 to Page 5, line 6).

The computer-implemented algorithm employed to streamline quotient ($Q = a_0a_1a_2....a_b$) calculations found by dividing dividend ($X = x_1x_2....x_a$) by divisor ($Y = y_1y_2....y_n$) proceeds as follows: (a) aligning the first non-zero bit of X with the first non-zero digit of Y; (b) defining a signed-digit partial remainder series R_i where $R_0 = Y$, a first sign series of the partial remainder S_i where $S_0 = 0$, a second sign series of the partial remainder S_ri, a quotient bit series a_i, and a counter i beginning from zero; (c) subtracting X from R_i which yields next signed-digit partial remainder R_{i+1}; (d) setting the sign of R_{i+1} to S_{ri+1}; (e) setting the result of exclusive-OR of S_i and S_{ri+1} to the true sign of the next remainder S_{i+1}; (f) setting a_i to 1 if $S_{i+1} = 0$ or $R_{i+1} = 0$; (g) setting a_i to 0 if $S_{i+1} = 1$; (h) inverting the signs of all digits of R_{i+1} if $S_{i+1} = 1$; (I) shifting R_{i+1} left by one bit; (j) adding 1 to i; and (k) repeating steps (c) to (j) until i reaches a predetermined value or $R_{i+1} = 0$. (See Specification, Page 5, line 8 to line 24).

VLSI (Very Large-Scale Integration) digital information processing systems can suitably implement this algorithm. In addition, this algorithm can be extended to be implemented in bit-parallel digital information processing systems. (See Specification, Page 17, line 17 to line 20).

2. Summary - Claim by Claim

Claim 1 is an independent method claim reciting specific steps for generating in a digital information processing system data representative of a quotient from data representative of a dividend and a divisor. To achieve this object, the system implements an algorithm comprising the following setup steps of aligning divisor and dividend system data and then defining a signed-digit partial remainder series where this series is initially set to be equal to the divisor, a first and a second sign series of the partial remainder where the first sign series is initially set to be equal to 0, a quotient bit series and a counter beginning with zero (references to "next" below refer to this counter + 1). The system-implemented algorithm then continues with the following data generation steps of subtracting the dividend data from the signed-digit partial remainder series to generate the next signed-digit partial remainder, setting this next signed-digit partial remainder to the next second sign series of the partial remainder, determining the result of exclusive-OR of the first and next second sign series of the partial remainder, and setting this result to the true sign of the next first sign series of the partial remainder, setting the quotient bit series to 1 if the next first sign series of the partial remainder is 0 or if the next signed-digit partial remainder is 0, setting the quotient bit series to 0 if the next first sign series of the partial remainder is 1, inverting the signs of all digits of the next signed-digit partial remainder if next first sign series of the partial remainder is 1, shifting the digits of the next signed-digit partial remainder left by one bit, adding one to the counter, and then repeating these data generation steps until the counter reaches a predetermined value or until the next signed-digit partial remainder is 0. The result of this system-implemented algorithm is then stored in a system memory as data representative of a quotient.

Claim 2 is an independent method claim reciting specific steps for generating in a digital information processing system data representative of a signed magnitude quotient from data representative of a signed dividend and a signed divisor. To achieve this object, the system implements an algorithm comprising the following setup steps of determining the result of exclusive-OR of the sign of signed divisor and signed dividend and setting the sign of the quotient equal to this result, defining a divisor, a dividend, a signed-digit partial remainder series where this series is initially set to be equal to the divisor, a first sign series of the partial remainder set initially to be equal to 0, a second sign series of the partial remainder, a quotient bit series, and a counter beginning with zero (references to "next" below refer to this counter + 1), and then aligning divisor and dividend system data. The system-implemented algorithm then continues with the following data generation steps of subtracting the dividend data from the signed-digit partial remainder series to generate the next signed-digit partial remainder, setting this next signed-digit partial remainder to the next second sign series of the partial remainder, determining the result of exclusive-OR of the first and next second sign series of the partial remainder, and setting this result to the true sign of the next first sign series of the partial remainder, setting the quotient bit series to 1 if the next first sign series of the partial remainder is 0 or if the next signed-digit partial remainder is 0, setting the quotient bit series to 0 if the next first sign series of the partial remainder is 1, inverting the signs of all digits of the next signed-digit partial remainder if next first sign series of the partial remainder is 1, shifting the digits of the next signed-digit partial remainder left by one bit, adding one to the counter, and then repeating these data generation steps until the counter reaches a predetermined value or until the next signed-digit partial remainder is 0. The result of this system-implemented algorithm is then stored in a system memory as data representative of a signed-magnitude quotient.

Claim 3 is an independent apparatus claim reciting a digital information processing system which generates data representative of a quotient from data representative of a divisor and a dividend. To achieve this object, the digital information processing system implements an algorithm reciting the following specific setup steps for

generating this data: aligning divisor and dividend system data and then defining a signed-digit partial remainder series where this series is initially set to be equal to the divisor, a first and a second sign series of the partial remainder where the first sign series is initially set to be equal to 0, a quotient bit series and a counter beginning with zero (references to "next" below refer to this counter + 1). The system-implemented algorithm then continues with the following data generation steps of subtracting the dividend data from the signed-digit partial remainder series to generate the next signed-digit partial remainder, setting this next signed-digit partial remainder to the next second sign series of the partial remainder, determining the result of exclusive-OR of the first and next second sign series of the partial remainder, and setting this result to the true sign of the next first sign series of the partial remainder, setting the quotient bit series to 1 if the next first sign series of the partial remainder is 0 or if the next signed-digit partial remainder is 0, setting the quotient bit series to 0 if the next first sign series of the partial remainder is 1, inverting the signs of all digits of the next signed-digit partial remainder if next first sign series of the partial remainder is 1, shifting the digits of the next signed-digit partial remainder left by one bit, adding one to the counter, and then repeating these data generation steps until the counter reaches a predetermined value or until the next signed-digit partial remainder is 0. The result of this system-implemented algorithm is then stored in a system memory as data representative of a quotient.

Claim 4 is an independent apparatus claim reciting a digital information processing system which generates data representative of a signed magnitude quotient from data representative of a signed divisor and a signed dividend. To achieve this object, the digital information processing system implements an algorithm reciting the following specific setup steps for generating this data: determining the result of exclusive-OR of the sign of signed divisor and signed dividend and setting the sign of the quotient equal to this result, defining a divisor, a dividend, a signed-digit partial remainder series where this series is initially set to be equal to the divisor, a first sign series of the partial remainder set initially to be equal to 0, a second sign series of the partial remainder, a quotient bit series, and a counter beginning with zero (references to

"next" below refer to this counter + 1), and then aligning divisor and dividend system data. The system-implemented algorithm then continues with the following data generation steps of subtracting the dividend data from the signed-digit partial remainder series to generate the next signed-digit partial remainder, setting this next signed-digit partial remainder to the next second sign series of the partial remainder, determining the result of exclusive-OR of the first and next second sign series of the partial remainder, and setting this result to the true sign of the next first sign series of the partial remainder, setting the quotient bit series to 1 if the next first sign series of the partial remainder is 0 or if the next signed-digit partial remainder is 0, setting the quotient bit series to 0 if the next first sign series of the partial remainder is 1, inverting the signs of all digits of the next signed-digit partial remainder if next first sign series of the partial remainder is 1, shifting the digits of the next signed-digit partial remainder left by one bit, adding one to the counter, and then repeating these data generation steps until the counter reaches a predetermined value or until the next signed-digit partial remainder is 0. The result of this system-implemented algorithm is then stored in a system memory as data representative of a signed-magnitude quotient.

D. ISSUES

The only issue presented in this appeal can concisely be stated to be as follows:

1. Whether Claims 1-4 are unpatentable under 35 USC 101 as being directed to non-statutory subject matter.

E. GROUPING OF THE CLAIMS

Appellant states that each one of the claims is separately patentable, and, if the ground of rejection is sustained as to any one of the rejected claims, it is submitted that

such rejection will not be equally applicable to all of the claims.

Specifically, it is submitted that Claims 1 and 2 are directed to methods for achieving the improvement described in the present invention. On the other hand, Claims 3 and 4 are apparatus claims drawn to systems which achieve the improvement described in the present invention.

Furthermore, it is submitted that method Claim 1 recites a patentably distinct feature or features different from method Claim 2. Also, apparatus Claim 3 recites a patentably distinct feature or features different from apparatus Claim 4.

F. ARGUMENT

1. Rejection under 35 USC 101

(a) The Examiner's Rejection

The Examiner has rejected Claims 1-4, stating that these claims are directed to non-statutory subject matter. In support of this rejection, the Examiner refers to the first office action of the parent application (USSN 08/188,068), wherein the Examiner stated that the same rejection was based on the two-step test for determining non-statutory subject matter given by In re Freeman 197 USPQ 464 (CCPA 1980), as modified by In re Walter 205 USPQ 397 (CCPA 1982) and In re Abele 214 USPQ 682 (CCPA 1982). Referring to this two-step test, the Examiner determined (1) that Claims 1-2 of the parent application (all claims as originally filed in the parent) directly recited a mathematical algorithm, and (2) that the claims as a whole preempt the algorithm because "rewriting claims 1 and 2 without the mathematical steps yields a non-statutory subject matter." The Examiner further stated that the preamble language of Claims 1 and 2: "[i]n a system for digital information processing" was only a field of use or desired end use limitation failing to transform the subject matter into statutory subject matter.

In addressing the Applicants' remarks and amendments in the parent application, the Examiner stated that the steps of "storing in said memory" (found in Claims 1-2 and in added Claims 3-4) represented insignificant and non-essential post-solution activity because stored data "is a mere number that is not used to refine or limit other process steps."

In the file wrapper continuation application (USSN 08/510,740), the Examiner repeated the remarks of the preceding paragraph and continued the rejection of Claims 1-4 on the same bases. The Examiner went on to state that Claims 1-4 and the disclosure are directed to the solution of a mathematical problem and "merely manipulat[e] an abstract idea or solv[e] a purely mathematical problem without any practical limitation". The Examiner then reaffirmed the rejection based on this conclusion.

Finally, in response to the Applicants' response to the final rejection of Claims 1-4 in the file wrapper continuation application, the Examiner noted that these claims are not analogous to the claims of In re Alappat 31 USPQ 2d 1545 because the step of storing mathematical operations in a memory does not create a new machine which presents patentable subject matter.

(b) Detailed Discussion and Argument

Applicants respectfully submit that Claims 1-4 are directed to statutory subject matter.

Neither computers nor computer programs have been precluded from patentability by either the Supreme Court or the Court of Appeals for the Federal Circuit (CAFC). Both courts have concluded that patentability is not precluded from inventions constituting the application of computer software to computer

systems. The present invention calls for a system and method for generating quotient data in a digital information processing system (a computer, for instance) by implementation of an algorithm on the digital information processing system. As such, this algorithm is a computer program on the most basic level, all computer programs presenting no more than a sequence of instructions or steps in solving a problem on a computer. The CAFC in its recent Alappat decision (see below) was presented with an application presenting the same issues faced in the Examiner's rejection of the present application. The decision of patentability in Alappat, with its category of claimed subject matter being indistinguishable from that of the present application, dictates an identical patentability holding for the claims of the present application.

35 USC 101 reads:

> Whoever invents or discovers any new and useful *process, machine, manufacture, or composition of matter, or any new and useful improvement thereof*, may obtain a patent therefor, subject to the conditions and requirements of this title. (Emphasis added).

Certain categories, however, are not entitled to patent protection. The Supreme Court in Diamond v. Diehr, 450 U.S. 175, 185, 209 USPQ 1 (1981) explained that there are three such categories: "laws of nature, natural phenomena, and abstract ideas." In addition, certain mathematical subject matter has also been held by the Supreme Court to be unpatentable because standing alone, such matter "represent[s] nothing more than *abstract ideas* until reduced to some type of practical application. . . ." In re Alappat, 31 USPQ2d 1545, 1556, citing and interpreting Diehr, Parker v. Flook, 437 U.S. 584, 198 USPQ 193, and Gottschalk v. Benson, 409 U.S. 63, 175 USPQ 548. However, "[i]t is well settled that a statutory invention will result from the *application* of a scientific truth (law of nature) to an otherwise statutory structure or process." In re Walter, 205 USPQ

397, 406 (CCPA 1980). Thus, subject to these few exceptions, 35 USC 101 is read broadly, in concert with the Congressional mandate that "any new and useful process, machine, manufacture, or composition of matter" is patentable.

The test to determine whether the application of a scientific truth (in the present case, an algorithm) to an otherwise statutory structure or process has been performed is two-part, and is defined in In re Freeman, 197 USPQ 464, 471 (CCPA 1980), as modified by In re Walter, and In re Abele, 214 USPQ 682 (CCPA 1982):

(1) determine whether the claim(s) directly or indirectly recite an algorithm, and if it does,

(2) the claim(s) as a whole must be analyzed to determine if the algorithm is "applied in any manner to physical elements or process steps," and not just circumscribed by a field of use limitation or non-essential post-solution activity. In re Abele at 686 (citations omitted).

As described in the Summary of the Invention (Section C above), this invention recites an algorithm and applies it to a digital information processing system (a computer, for instance) in order to accelerate the system's quotient operations.

Appellants' amended independent Claim 1 calls for . . . [i]n a system for digital information processing, said system having a memory, a method for generating data representative of a quotient $Q = a_0a_1a_2....a_b$ from data representative of a divisor $Y = y_1y_2....y_n$ and data representative of a dividend $X = x_1x_2....x_a$ comprising the steps of:. . . . Later in Claim 1, the method calls for . . . storing in said memory as said data representative of a quotient, a quotient resulting . . .

The present invention is clearly restricted to the implementation of a specific algorithm in digital information processing systems (such as computers) - thereby

being defined as a program: "[i]n computer practice, a detailed sequence of instructions representing an algorithm (the necessary steps in solving a problem) that can be implemented by a computer." *The Illustrated Dictionary of Electronics*, 6th Ed., p. 522 (1994).

Computer systems being in and of themselves patentable subject matter, the Court of Customs and Patent Appeals has carefully noted that

> ". . . no basis exists for a moratorium on protection of inventions embodying or using computer programs. Such broad prohibition could subject meritorious statutory inventions to unabatable piracy, and could forestall invention disclosure, the hallmark of the patent system, until Congress chooses to act." In re de Castelet, 195 USPQ 439, 443.

The Supreme Court has also been careful to avoid foreclosing patentability toward computer programs, saying in Benson at 71: "[i]t is said that the decision precludes a patent for any program servicing a computer. We do not so hold."

Indeed, the Supreme Court's decision in Diehr was recently cited by the CAFC in Alappat (at 1557), a case presenting issues very similar to the issues now being faced in the present appeal. The CAFC's bracketed insertions into its citation of Diehr are instructive here:

> "[w]hen a claim containing a mathematical formula [, mathematical algorithm, or the like,] implements or applies that formula [, equation, algorithm , or the like,] in a structure or process which, when considered as a whole, is performing a function which patent laws were designed to protect (e.g., transforming or reducing an article to a different state or thing), then the claim satisfies the requirements of § 101." (Emphasis omitted).

Once again, Claim 1 calls for "a method of generating data representative of a quotient" "in a system for digital information processing, said system having a memory. . . ."

The Applicant submits that the present invention contains an algorithm, implements that algorithm in a digital information processing system, and works to improve the functioning of the digital information processing system by speeding and increasing the efficiency with which the system performs quotient calculations. Application of an algorithm to a system, thereby improving the system, is an invention that § 101 was intended to protect. The Supreme Court in Diehr pointed this out clearly: "[i]t is now commonplace that applications of a law of nature or mathematical formula to a known structure or process may well be deserving of patent protection." Diehr at 209.

In Alappat, the CAFC faced an application which, for all intents and purposes, presented identical issues as those being faced in the present application. While the Court noted that the independent claim 15 of Alappat, "may be viewed as a series of mathematical calculations" (Alappat at 1558), the Court nevertheless found claim 15 of Alappat to be directed to patentable subject matter. Claim 15 recited four means elements: 2 arithmetic logic circuits, a pair of barrel shifters, and a memory (Alappat at 1555) for transforming data from one form to another. This claim has a mirror image in Claim 1 of the present application. In Claim 1, a mathematical algorithm is implemented on a digital information processing system (a computer, for instance, illustrated further in Figure 1 as including an arithmetic logic unit), which includes a memory. Just as Alappat implements a series of mathematical steps to transform data from one form to another, the present invention describes a method which implements an algorithm on a digital information processing system to transform data from one form (divisor and dividend data) to another (quotient data). Just as the data

resulting from Alappat's steps is useful to produce a smooth waveform on a cathode-ray tube (CRT), the present invention's data resulting from implementation of an algorithm on a digital information processing system is useful to increase the processing speed of a computer performing quotient calculations.

The CAFC in Alappat recognized that the steps performed by the elements in Alappat's application could be performed and would read on a general purpose computer programmed to carry out the claimed invention. Such is also the case with the present invention. However, this fact did not stop the Court from finding patentable subject matter in Alappat's claims. "We have held that such programming creates a new machine, because a general purpose computer in effect becomes a special purpose computer once it is programmed to perform particular functions pursuant to instructions from program software." Alappat at 1558. Such is enough, according to the CAFC, to preclude a finding of unpatentable subject matter. In like manner, when a digital information processing system (such as a computer) is directed to implement the algorithm included in the claims of the present invention, the *system itself* is considered to be changed, creating statutory subject matter for patenting.

The CAFC went on to note that the opposite could not be the case. "Under the Board's reasoning, a programmed general purpose computer could never be viewed as patentable subject matter under § 101. This reasoning is without basis in law. The Supreme Court has never held that a programmed computer may never be entitled to patent protection." (*Id.* at 1558). To hold that the system and methods claimed in the present invention are directed to non-statutory subject matter would effectively bar protection to any digital information processing system or method having inventive steps or elements not in hardware or structure, but in operation and instruction. By logical extension, this would bar patent protection to any such systems which utilize known structure and

hardware, yet whose inventive step(s) or element(s) resides in the manner in which the system and hardware operate and are instructed. As just cited, this is clearly not what was intended by the CAFC or the Supreme Court.

Claim 1 of the present invention claims a method for finding a quotient in a digital information processing system such as a computer. This claim describes doing so by calling for a digital information processing system to carry out a recited algorithm. Claim 1 cannot be infringed by mere mental steps or paper and pencil. Claim 1 only reads on a process carried out in a digital information processing system, which is a type of machine. It improves the machine and it is certainly the type of technology which cries out for patent protection.

Applying the same logic and reasoning as the CAFC did in Alappat, the Applicant submits that there is no reason why a different conclusion as to patentable subject matter should be reached in the present application. The similarity between the Claim 1 of the present invention and Claim 15 of Alappat's application, along with the similarity between the subject matter of the claims in both applications, leads to the conclusion that the claims of the present invention are directed to statutory subject matter as required in 35 USC 101.

Claim 2 of the present application also calls for "a method for generating data representative of a . . . quotient" "in a system for digital information processing, said system having a memory. . . ." Claim 2 further includes the ability to generate "signed magnitude quotient" data. (Emphasis added).

The arguments set forth above for Claim 1 similarly apply to Claim 2. It is respectfully submitted that Claim 2 is also not directed to non-statutory subject matter.

Claim 3 of the present application calls for "a system for digital information processing, said system having a memory for storing data, including data representative of a quotient . . . said data representative of a quotient generated by a method comprising the steps of: . . . ." (Emphasis added).

The arguments set forth above for Claim 1 similarly apply to Claim 3. Claim 3 is directed to a machine. Machines clearly fall within the scope of 35 USC 101. It is respectfully submitted that Claim 3 is also not directed to non-statutory subject matter.

Claim 4 of the present application calls for "a system for digital information processing, said system having a memory for storing data, including data representative of a signed magnitude quotient . . . said data representative of a signed magnitude quotient generated by a method comprising the steps of: . . . ." (Emphasis added).

The arguments set forth above for Claim 1 similarly apply to Claim 4. Claim 4 is directed to a machine. Machines clearly fall within the scope of 35 USC 101. It is respectfully submitted that Claim 4 is also not directed to non-statutory subject matter.

As presented in the above arguments, the claims of the present invention are directed to a digital processing system and methods in a digital processing system for generating data representative of a quotient from data representative of a divisor and a dividend. None of the claims are directed toward an abstract idea, principle, or law of nature. None of the claims can be infringed by method steps or by paper and pencil. Instead, all claims describe a very real and concrete digital processing system implementing a detailed sequence of instructions representing an algorithm; in essence, a digital information processing system implementing a program. Both the Supreme Court and the CAFC have been careful to note that

computer systems, computer programs, and computer systems implementing computer programs are not precluded from the intended scope of patentability covered by 35 USC 101. The patentable claims of In re Alappat are a prime example of this principle's application: a series of admittedly mathematical steps to transform data from one form to another on a computer system. No significant differences between this Alappat system and the systems and methods of the present invention justify a finding of unpatentability for the claims presented in the present application. As the Court stated in Alappat, "a computer operating pursuant to software *may* represent patentable subject matter, provided, of course, that the claimed subject matter meets all of the other requirements of Title 35." (Alappat at 1558). In the present application, the Examiner does not contend that the claimed subject matter fails to meet any other requirement of Title 35. The sole rejection is based on 35 USC 101 and that rejection must fall in concert with the Court's decision in Alappat.

(d) Appellant's Request

The Board is respectfully requested to reverse the Examiner's rejection of Claims 1-4 under 35 USC 101.

G. CONCLUSION

The Appellant has shown above that each of the Examiner's objections and rejections are unfounded. Therefore, the Board is requested to reverse: the rejection of Claims 1-4 under 35 USC 101.

The Commissioner is authorized to charge any additional fees which may be required or credit overpayment to deposit account 12-0415 and, in particular, if this response is not timely filed, then the Commissioner is authorized to treat this response as including a petition to extend the time period pursuant to 37 CFR 1.136(a) requesting

an extension of time of the number of months necessary to make this response timely filed and the petition fee due in connection therewith may be charged to deposit account no. 12-0415.

Respectfully submitted,
LADAS & PARRY

By _____
Richard J. Paciulan
Reg. No. 28,248
Attorney for Assignee

Ladas & Parry
5670 Wilshire Boulevard, Suite 2100
Los Angeles, CA 90036
(213) 934-2300

APPENDIX

The Claims involved in the Appeal are as follows:

1. In a system for digital information processing, said system having a memory, a method for generating data representative of a quotient $Q = a_0a_1a_2....a_b$ from data representative of a divisor $Y = y_1y_2....y_n$ and data representative of a dividend $X = x_1x_2....x_a$, comprising the steps of:

   (a) aligning the first non-zero bit of X with the first non-zero digit of Y;

   (b) defining a signed-digit partial remainder series R_i where $R_0 = Y$, a first sign series of the partial remainder S_i where $S_0 = 0$, a second sign series of the partial remainder S_{ri}, a quotient bit series a_i, and a counter i beginning from zero;

   (c) subtracting X from R_i which yields next signed-digit partial remainder R_{i+1};

   (d) setting the sign of R_{i+1} to S_{ri+1};

   (e) setting the result of exclusive-OR of S_i and S_{ri+1} to the true sign of the next remainder S_{i+1};

   (f) setting a_i to 1 if $S_{i+1} = 0$ or $R_{i+1} = 0$;

   (g) setting a_i to 0 if $S_{i+1} = 1$;

   (h) inverting the signs of all digits of R_{i+1} if $S_{i+1} = 1$;

   (i) shift R_{i+1} left by one bit;

(j) adding 1 to i;

(k) repeating steps (c) to (j) until i reaches a predetermined value or $R_{i+1} = 0$; and

(l) storing in said memory as said data representative of a quotient, a quotient resulting from step (k).

2. In a system for digital information processing, said system having a memory for storing data , a method for generating data representative of a signed magnitude quotient $Q_2 = a_s a_0.a_1 a_2....a_b$ from data representative of a signed divisor $Y_s = y_s.y_1 y_2....y_n$, , and data representative of a signed dividend $X_s = x_s.x_1 x_2....x_s$, comprising the steps of:

(a) obtaining a_s from the result of exclusive-OR of y_s and x_s;

(b) defining a divisor $Y = y_1 y_2....y_n$, a dividend $X = x_1 x_2....x_s$, a signed-digit partial remainder series R_i where $R_0 = Y$, a first sign series of the partial remainder S_i where $S_0 = 0$, a second sign series of the partial remainder S_{ri}, a quotient bit series a_i, and a counter i beginning from zero;

(c) aligning the first non-zero bit of X with the first non-zero digit of Y;

(d) subtracting X from R_i which yields next signed-digit partial remainder R_{i+1};

(e) setting the sign of R_{i+1} to S_{ri+1};

(f) setting the result of exclusive-OR of S_i and S_{ri+1} to the true sign of the next remainder S_{i+1};

(g) setting a_i to 1 if $S_{i+1} = 0$ or $R_{i+1} = 0$;

(h) setting a_i to 0 if $S_{i+1} = 1$;

(i) inverting the signs of all digits of R_{i+1} if $S_{i+1} = 1$;

(j) shift R_{i+1} left by one bit;

(k) adding 1 to i; and

(l) repeating steps (d) to (k) until i reaches a predetermined value or $R_{i+1} = 0$; and

(l) storing in said memory as said data representative of a signed magnitude quotient, a quotient resulting from step (k).

3. A system for digital information processing, said system having a memory for storing data, including data representative of a quotient $Q = a_0a_1a_2....a_b$ from data representative of a divisor $Y = y_1y_2....y_n$ and data representative of a dividend $X = x_1x_2....x_a$, said data representative of a quotient generated by a method comprising the steps of:

(a) aligning the first non-zero bit of X with the first non-zero digit of Y;

(b) defining a signed-digit partial remainder series R_i where $R_0 = Y$, a first sign series of the partial remainder S_i where $S_0 = 0$, a second sign series of the partial remainder S_{ri}, a quotient bit series a_i, and a counter i beginning from zero;

(c) subtracting X from R_i which yields next signed-digit partial remainder R_{i+1};

(d) setting the sign of R_{i+1} to S_{ri+1};

(e) setting the result of exclusive-OR of S_i and S_{ri+1} to the true sign of the next remainder S_{i+1};

(f) setting a_i to 1 if $S_{i+1} = 0$ or $R_{i+1} = 0$;

(g) setting a_i to 0 if $S_{i+1} = 1$;

(h) inverting the signs of all digits of R_{i+1} if $S_{i+1} = 1$;

(i) shift R_{i+1} left by one bit;

(j) adding 1 to i;

(k) repeating steps (c) to (j) until i reaches a predetermined value or $R_{i+1} = 0$; and

(l) storing in said memory as said data representative of a quotient , a quotient resulting from step (k).

4. A system for digital information processing, said system having a memory for storing data, including data representative of a signed magnitude quotient $Q_2 = a_s a_0.a_1 a_2....a_b$ from data representative of a signed divisor $Y_s = y_s.y_1 y_2....y_n$, , and data representative of a signed dividend $X_s = x_s.x_1 x_2....x_s$, said data representative of a signed magnitude quotient generated by a method comprising the steps of:

(a) obtaining a_s from the result of exclusive-OR of y_s and x_s;

(b) defining a divisor $Y = y_1 y_2....y_n$, a dividend $X = x_1 x_2....x_s$, a signed-digit partial remainder series R_i where $R_0 = Y$, a first sign series of the partial remainder S_i where $S_0 = 0$, a second sign series of the partial remainder S_{ri}, a quotient bit series a_i, and a counter i beginning from zero;

(c) aligning the first non-zero bit of X with the first non-zero digit of Y;

(d) subtracting X from R_i which yields next signed-digit partial remainder R_{i+1};

(e) setting the sign of R_{i+1} to S_{ri+1};

(f) setting the result of exclusive-OR of S_i and S_{ri+1} to the true sign of the next remainder S_{i+1};

(g) setting a_i to 1 if $S_{i+1} = 0$ or $R_{i+1} = 0$;

(h) setting a_i to 0 if $S_{i+1} = 1$;

(i) inverting the signs of all digits of R_{i+1} if $S_{i+1} = 1$;

(j) shift R_{i+1} left by one bit;

(k) adding 1 to i;

(l) repeating steps (d) to (k) until i reaches a predetermined value or $R_{i+1} = 0$; and

(m) storing in said memory as said data representative of a signed magnitude quotient, a quotient resulting from step (l).